



This is part of **Family API** which allow to create dual-os version of program runs under OS/2 and DOS

**Note:** This is legacy API call. It is recommended to use 32-bit equivalent

2021/09/17 04:47 · prokushev · [0 Comments](#)

2021/08/20 03:18 · prokushev · [0 Comments](#)

# DosSetDateTime

This call is used to set the date and time that are maintained by the operating system.

## Syntax

```
DosSetDateTime (DateTime)
```

## Parameters

- DateTime ([PDATETIME](#)) - input : Address of the date and time structure:
  - hours ([UCHAR](#)) - Current hour (0-23)
  - minutes ([UCHAR](#)) - Current minute (0-59)
  - seconds ([UCHAR](#)) - Current second (0-59)
  - hundredths ([UCHAR](#)) - Current hundredth of a second (0-99)
  - day ([UCHAR](#)) - Current day (1-31)
  - month ([UCHAR](#)) - Current month (1-12)
  - year ([USHORT](#)) - Current year (1980-2079)
  - timezone ([SHORT](#)) - Minutes west of UTC (Universal Time Coordinate -720 to 720).
  - weekday ([UCHAR](#)) - Current day of the week. This value is ignored and is calculated from the other parameter information.

## Return Code

rc ([USHORT](#)) - return

Return code descriptions are:

- 0 NO\_ERROR
- 327 ERROR\_TS\_DATETIME

## Remarks

The value of `timezone` is the difference in minutes between the current time zone and UTC. This is a positive number if earlier than UTC, and negative number if it is later. For Eastern Standard Time this value is 300 (five hours earlier than UTC).

To get the date and time, issue [DosGetDateTime](#). If the application is executing in the OS/2 environment, it is more efficient to obtain these variables by calling [DosGetInfoSeg](#) instead of this function. However, applications written to the family API cannot depend on the availability of [DosGetInfoSeg](#).

Not adhering to the limits on any of the parameters results in the return code being set to `rc = 327` (`ERROR_TS_DATETIME`). Also, OS/2 verifies that the day is possible for the month and the year (even for leap year). If the day is not reasonable, OS/2 will also set `rc = 327`.

## Example Code

### C Binding

```
typedef struct _DATETIME { /* date */
    UCHAR    hours;        /* current hour */
    UCHAR    minutes;      /* current minute */
    UCHAR    seconds;      /* current second */
    UCHAR    hundredths;   /* current hundredths of a second */
    UCHAR    day;          /* current day */
    UCHAR    month;        /* current month */
    USHORT   year;         /* current year */
    SHORT    timezone;     /* minutes of time west of UTC */
    UCHAR    weekday;      /* current day of week */
} DATETIME;

#define INCL_DOSDATETIME

USHORT    rc = DosSetDateTime(DateTime);
PDATETIME DateTime; /* Date/time structure */
USHORT    rc;        /* return code */
```

The following example obtains and prints date and time information. It then changes the system date to 5/10/1987 and prints the updated information.

```
#define INCL_DOSDATETIME
#include <os2.h>

main()
{
    DATETIME    DateTime; /* Structure to hold date/time info. */
    USHORT      rc;
```

```

rc = DosGetDateTime(&DateTime);      /* Address of d/t structure */
printf("Today is %d-%d-%d; the time is %d:%d\n", DateTime.month,
       DateTime.day, DateTime.year, DateTime.hours, DateTime.minutes);
DateTime.day = 10;
DateTime.month = 5;
DateTime.year = 1987;
printf("The new date is %d-%d-%d; the time is %d:%d\n", DateTime.month,
       DateTime.day, DateTime.year, DateTime.hours, DateTime.minutes);
rc = DosSetDateTime(&DateTime);      /* Address of d/t structure */
printf("rc is %d\n", rc);
}

```

## MASM Binding

```

DATETIME struc
    date_hours      db  ? ;current hour
    date_minutes    db  ? ;current minute
    date_seconds     db  ? ;current second
    date_hundredths db  ? ;current hundredths of a second
    date_day        db  ? ;current day
    date_month       db  ? ;current month
    date_year        dw  ? ;current year
    date_timezone    dw  ? ;minutes of time west of UTC
    date_weekday     db  ? ;current day of week
DATETIME ends

EXTRN DosSetDateTime:FAR
INCL_DOSDATETIME EQU 1

PUSH@ OTHER DateTime      ;Date/time structure
CALL DosSetDateTime

```

Returns WORD

## Note

Text based on [http://www.edm2.com/index.php/DosSetDateTime\\_\(Legacy\)](http://www.edm2.com/index.php/DosSetDateTime_(Legacy))

Family API		
DOS	Process Manager	<a href="#">DosBeep</a> <a href="#">DosExit</a> <a href="#">DosSleep</a> <a href="#">DosExecPgm</a>
	File Manager	<a href="#">DosChDir</a> <a href="#">DosChgFilePtr</a> <a href="#">DosClose</a> <a href="#">DosDelete</a> <a href="#">DosDupHandle</a> <a href="#">DosMkDir</a> <a href="#">DosMove</a> <a href="#">DosQCurDir</a> <a href="#">DosQCurDisk</a> <a href="#">DosSetFileMode</a> <a href="#">DosOpen</a> <a href="#">DosQFileInfo</a> <a href="#">DosRead</a> <a href="#">DosQFileMode</a> <a href="#">DosQFSInfo</a> <a href="#">DosQVerify</a> <a href="#">DosRmDir</a> <a href="#">DosSelectDisk</a> <a href="#">DosFindClose</a> <a href="#">DosFindFirst</a> <a href="#">DosFindNext</a> <a href="#">DosSetFileInfo</a> <a href="#">DosSetVerify</a> <a href="#">DosWrite</a> <a href="#">DosFileLocks</a> <a href="#">DosSetFHandState</a> <a href="#">DosNewSize</a> <a href="#">DosBufReset</a> <a href="#">DosQFHandState</a> <a href="#">DosSetFSInfo</a>
	Memory Manager	<a href="#">DosFreeSeg</a> <a href="#">DosSubAlloc</a> <a href="#">DosSubFree</a> <a href="#">DosSubSet</a> <a href="#">DosAllocHuge</a> <a href="#">DosAllocSeg</a> <a href="#">DosReallocHuge</a> <a href="#">DosReallocSeg</a> <a href="#">DosGetHugeShift</a> <a href="#">DosCreateCSAlias</a>
	NLS	<a href="#">DosCaseMap</a> <a href="#">DosGetCtryInfo</a> <a href="#">DosGetDBCSEv</a> <a href="#">DosSetCtryCode</a> <a href="#">DosGetCollate</a> <a href="#">DosGetMessage</a> <a href="#">DosInsMessage</a> <a href="#">DosPutMessage</a>
	Date and Time	<a href="#">DosSetDateTime</a> <a href="#">DosGetDateTime</a>
	Devices	<a href="#">DosDevConfig</a> <a href="#">DosDevIOCtl</a> <a href="#">DosDevIOCtl2</a>
	Signals	<a href="#">DosHoldSignal</a> <a href="#">DosSetSigHandler</a>
	Misc	<a href="#">BadDynLink</a> <a href="#">DosGetEnv</a> <a href="#">DosGetMachineMode</a> <a href="#">DosGetVersion</a> <a href="#">DosError</a> <a href="#">DosErrClass</a> <a href="#">DosSetVec</a>
KBD		<a href="#">KbdCharIn</a> <a href="#">KbdFlushBuffer</a> <a href="#">KbdGetStatus</a> <a href="#">KbdSetStatus</a> <a href="#">KbdStringIn</a> <a href="#">KbdPeek</a>
VIO		<a href="#">VioGetBuf</a> <a href="#">VioGetConfig</a> <a href="#">VioGetCurPos</a> <a href="#">VioGetCurType</a> <a href="#">VioGetPhysBuf</a> <a href="#">VioReadCellStr</a> <a href="#">VioReadCharStr</a> <a href="#">VioScrollUp</a> <a href="#">VioScrollDn</a> <a href="#">VioScrollLf</a> <a href="#">VioScrollRt</a> <a href="#">VioScrUnLock</a> <a href="#">VioSetCurPos</a> <a href="#">VioSetCurType</a> <a href="#">VioSetMode</a> <a href="#">VioGetMode</a> <a href="#">VioShowBuf</a> <a href="#">VioWrtCellStr</a> <a href="#">VioWrtCharStr</a> <a href="#">VioWrtCharStrAtt</a> <a href="#">VioWrtNAttr</a> <a href="#">VioWrtNCell</a> <a href="#">VioWrtNChar</a> <a href="#">VioWrtTTY</a> <a href="#">VioScrLock</a> <a href="#">VioPopUp</a>
Tools		<a href="#">BIND</a>
Modules		<a href="#">DOSCALLS.DLL</a> <a href="#">VIOCALLS.DLL</a> <a href="#">KBDCALLS.DLL</a> <a href="#">MSG.DLL</a>
Libraries		<a href="#">API.LIB</a> <a href="#">OS2386.LIB</a> <a href="#">FAPI.LIB</a> <a href="#">DOSCALLS.LIB</a> <a href="#">SUBCALLS.LIB</a>

2018/08/25 15:05 · [prokushev](#) · [0 Comments](#)

From:  
<http://ftp.osfree.org/doku/> - **osFree wiki**

Permanent link:  
<http://ftp.osfree.org/doku/doku.php?id=en:docs:fapi:dossetdatetime>

Last update: **2021/12/05 11:01**

